View-agnostic Human Exercise Cataloging with Single MmWave Radar

ALAN LIU, Georgia Institute of Technology, USA
YU-TAI LIN, Georgia Institute of Technology, USA
KARTHIKEYAN SUNDARESAN, Georgia Institute of Technology, USA

Advances in mmWave-based sensing have enabled a privacy-friendly approach to pose and gesture recognition. Yet, providing robustness with the sparsity of reflected signals has been a long-standing challenge towards its practical deployment, constraining subjects to often face the radar. We present *RF-HAC*– a first-of-its-kind system that brings robust, automated and real-time human activity cataloging to practice by not only classifying exercises performed by subjects in their natural environments and poses, but also tracking the corresponding number of exercise repetitions. *RF-HAC*'s unique approach (i) brings the diversity of multiple radars to scalably train a novel, *self-supervised, pose-agnostic* transformer-based exercise classifier directly on 3D RF point clouds with minimal manual effort and be deployed on a single radar; and (ii) leverages the underlying doppler behavior of exercises to design a robust *self-similarity based segmentation* algorithm for counting the repetitions in unstructured RF point clouds. Evaluations on a comprehensive set of challenging exercises in both seen and unseen environments/subjects highlight *RF-HAC*'s robustness with high accuracy (over 90%) and readiness for real-time, practical deployments over prior art.

CCS Concepts: • **Human-centered computing** → **Ubiquitous and mobile computing**; *Human computer interaction (HCI)*; • **Computing methodologies** → *Semi-supervised learning settings.*

Additional Key Words and Phrases: mmWave Sensing, self-supervised learning, view-agnostic sensing, human activity recognition

ACM Reference Format:
Alan Liu, Yu-Tai Lin, and Karthikeyan Sundaresan. 2024. View-agnostic Human Exercise Cataloging with Single MmWave Radar. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 3, Article 117 (September 2024), 23 pages. https://doi.org/10.1145/3678512

1 INTRODUCTION

Physical activity has always been an integral part of human well-being. The recent pandemic has fueled the growing interest in remote health monitoring solutions from sleep monitoring to rehabilitation assistance. In particular, the ability to monitor and catalogue exercises autonomously, as shown in Fig. 1, enables analytics over time. The derived insights and recommendations for tuning the activity regimen, in turn enables several applications of automated exercise monitoring in personal well-being, assisted/elderly care, virtual gym assistance/log and physical therapy assistance/log, to name a few.

There are two main aspects to exercise cataloging: exercise classification and repetition count. Numerous solutions address different aspects of physical activity monitoring such as gait analysis [9, 44], vitals monitoring [49], sleep analysis [53], etc. Recent advances in computer vision, namely action recognition [7, 10] can serve as an effective modality in identifying actions and gestures, although at the expense of user privacy.

Authors' addresses: Alan Liu, Georgia Institute of Technology, USA, alanliu2@gatech.edu; Yu-Tai Lin, Georgia Institute of Technology, USA, ytlin1993@gatech.edu; Karthikeyan Sundaresan, Georgia Institute of Technology, USA, karthik@ece.gatech.edu.

2474-9567/2024/9-ART117
https://doi.org/10.1145/3678512

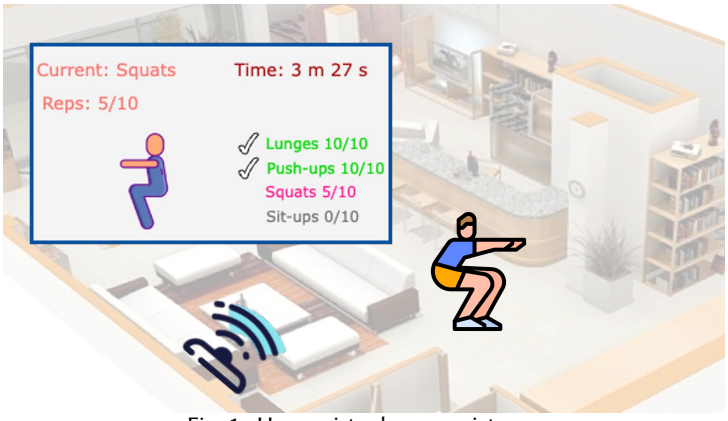


Fig. 1. Home virtual gym assistance

In contrast, RF-sensing based approaches, especially those involving higher mmWave frequencies (owing to higher bandwidth/resolution) are becoming popular both in academic research [50] as well as in the industry (e.g. Amazon Halo Rise [3]). Further, they are likely to be integrated with communication in our indoor WiFi access points in the near future. Such mmWave sensing solutions often involve an ML-driven approach to map the complex target-reflected RF signals to appropriate application-specific features needed for classification, without the strong implications for privacy. However, lacking in robustness for operation in everyday environments, they are yet to see wide-spread adoption.

Delivering a practical, automated human activity catologuer (HAC) poses three technical challenges, where RF-sensing based solutions fall short. (i) *robustness*: While mmWave radars provide high bandwidth and hence the needed resolution for sensing, their sparse reflections from targets make them highly sensitive to the target's orientation (relative to radar). This makes them suffer in performance by as much as 50% (seen in Section 3), when the target's orientation varies considerably from the one the system was trained for; furthermore, use of limited information such as 2D spectrograms (compared to 3D point clouds) also impacts the ability to accurately discriminate exercises in different orientations. (ii) *deployment overhead*: A potential solution is to train the system with data from many different target orientations. However, this incurs a significant amount of effort both for the target and data collection, not to mention the human/manual labeling of views and the associated additional training [48]. It still does not guarantee robustness to target orientations unseen in training. (iii) *automation*: Finally, beyond activity classification, an important role of a cataloguer is to also count the number of gestures/actions repeated by the target. However, ML-based approaches typically can classify an 'instance' of an exercise [42] for trained target orientations but are unable to automatically and dynamically segment a sequence of exercise repetitions in real-time depending on the dynamic speed of the exercise. The difficulty of this challenge is highlighted by a modest 65% accuracy achieved by a state-of-the-art ML-based segmentation approach [16] even on 2D videos. Further, enabling an ML approach to automatically segment a series of exercise repetitions, especially based on 3D RF data, would incur a large training overhead owing to the need for manual segmentation of the sensor data stream for generating ground truth [27].

Towards addressing these challenges, we propose *RF-HAC*[1], a novel mmWave-based automated, real-time activity cataloging system that can robustly classify exercises as well as count their repetitions for targets in real-time in everyday environments, by directly operating on 3D RF point clouds. *RF-HAC*'s novel design significantly advances state-of-the-art in two important directions:

[1]RF-HAC stands for RF-driven Human Activity Cataloguer

(i) a *self-supervised, view-agnostic exercise classification model for RF point clouds* that brings the diversity benefits of multiple radars to robustly classify various exercises (involving both upper and lower body) in practical single radar deployments with very little training overhead. *RF-HAC* avoids the large data collection overhead associated with numerous target orientations, by allowing targets to perform the exercise in a single orientation during training, while deploying multiple (three in our case) radars, uniformly spaced in a circle around the target, whose aggregated 360 degree RF point clouds are flexibly rotated to later obtain diverse perspectives on the RF features. It eliminates the need for manual labeling of data, which is complemented with a novel contrastive learning approach that enables robust view-agnostic classification – the model is trained by pitting the different radar views against each other through a mutual information based loss function on the easily composed contrastive (positive - e.g. same exercise with different orientations, negative - e.g. different exercises) samples, allowing it to learn only the exercise-differentiating features from the 3D RF point clouds without carrying any artifacts specific to the poses. Once trained, the model is ready for operation with a *single* radar, where the diversity gain of multiple radars has been embodied into the model's robustness to varying target orientations.

(ii) a *self-similarity based, scalable segmentation algorithm for unstructured RF point clouds* that brings automated repetition counting to dynamic RF point clouds. *RF-HAC* leverages the high correlation between proximal (spatially) points in a target's point cloud, to interpolate the doppler values of the unstructured point cloud and map it to a structured grid-based point cloud. The latter is used to analyze the self-similarity features of a 3D doppler point cloud and hence determine the appropriate segmentation of the data stream, leading to its repetition count. To overcome the high computational complexity of generating 3D self-similarity matrices (SSM) that stifle real-time operation, *RF-HAC* decomposes the problem into generation of three 2D SSMs corresponding to each of the planes (xy, yz, xz by mean pooling the third dimension) and aggregating them. We show that the decomposed version yields a performance very close to that of the 3D SSM, while avoiding its complexity to enable real-time performance.

RF-HAC's segmentation and classification modules synergistically reinforce each other to deliver a robust solution. The incoming sensor stream is segmented accurately as it arrives and fed as individual exercise instances, which helps further boost the classification model's accuracy by eliminating spurious multi-count and partial exercise segments. On the other hand, the model's identification of the exercise is further used to adjust the parameters of the segmentation algorithm to further increase its accuracy, especially for exercises that involve alternating limbs and result in multiple/harmonic segmented solutions.

RF-HAC is trained with three 60 GHz commodity radars each placed 120 degrees apart. However, it is deployed with only a single radar in real-world with practically no deployment overhead. While *RF-HAC* is built as an add-on processing module that receives 3D RF point clouds from the commodity radar to enable real-time exercise cataloging, it can easily be integrated with the radar's processing from a deployment perspective. Our comprehensive evaluations on a dictionary of 9 challenging exercises (Fig. 2a) that span both upper and lower body, reveal that *RF-HAC* delivers an accuracy of 93% in exercise classification and 94% in repetition counting to enable cataloging in practice. With no current repetition counting schemes, this is a gain of 13% over state-of-the-art models leveraging radar diversity (50% sans diversity) for classification and can be as high as 25-30% on challenging exercises operating close to the ground. *RF-HAC*'s ability to learn pose-agnostic features from radar diversity delivers robustness across subject orientations even for unseen environments and subjects, while suffering the least accuracy drop in partially occluded scenarios. Our contributions in this work can be summarized as follows:

• We propose a novel view-agnostic self-supervised model that embodies radar diversity to bring robustness to activity classification in everyday environments.

• We propose a light-weight, resilient segmentation algorithm for unstructured 3D RF point clouds that leverages self-similarity to enable automated repetition counting.

• We will open source a rich, multi-radar dataset for a dictionary of nine exercises in various environments that will be made publicly available to facilitate further research.

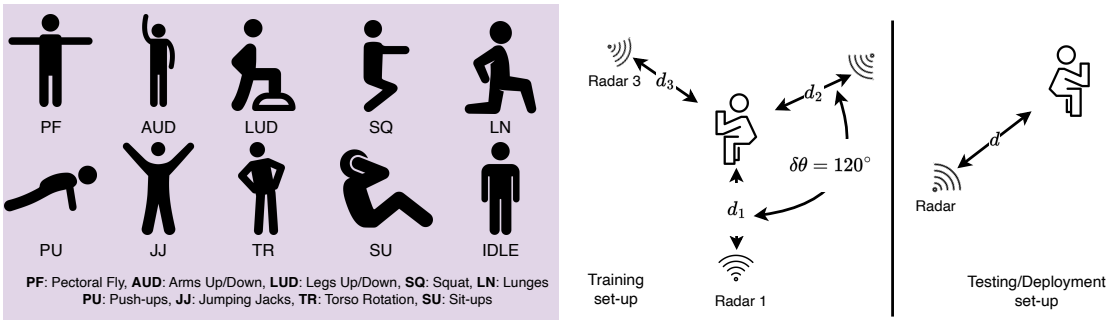
We believe *RF-HAC* is one of the first practical RF systems to provide an accurate, robust and automated exercise cataloging in real-time that opens the door for numerous applications in self and remote health management. While recent works have considered mmWave-based Human Activity Recognition, they lack several features required for practical, robust and flexible deployment. They either incur a large deployment overhead [35], lack view-agnostic recognition [37, 38, 40, 42, 48], and/or require additional sensor modes for training [25, 31]. Additionally, to the best of our knowledge, we are one of the first works to feature a mmWave-based exercise repetition counting system that along with recognition, is essential to deliver exercise cataloging.

2 BACKGROUND AND RELATED WORK

Human Activity Recognition has long been a problem in the field of computer vision [11, 13, 30, 32, 33]. One of its challenging problems includes pose recognition, especially when subjects were not facing the camera. Some recent works have approached this problem of view-agnostic pose estimation through the use of contrastive learning [36, 45, 52] on individual images. Another problem explored recently is that of repetition counting, which tries to find the number of times a repetition occurs given a video with repeating action using deep learning models [16] to only modest accuracies of 65-70%, highlighting the problem's challenging nature. However, with the focus being on 2d images or sequences of 2d images from cameras predominantly, 3d point cloud sequences have not been explored often.

RF sensors such as mmWave radars, which can produce 3d point cloud sequences, have become a promising alternative to computer vision for many applications including street view detection [20], vital signs monitoring [8, 21, 43], and human activity recognition [25, 38, 40, 46, 47] among others [12, 14, 19, 28, 39, 41], not only because of their lower cost, but also because of higher resolution and better privacy-sensitivity [50]. Most of these works employ a single radar chip that offers sparse information, restricting analysis to 1D or 2D range-doppler signals. Even the few working with point clouds [35, 37, 42, 48], do so with a sparse point cloud. Operating with inherently limited information, existing works with mmWave radar approaches [38, 40, 42, 47, 48] also require users to be facing the radar , restricting practical deployments. Some of the activity recognition works [25] incorporate vision for training, and are able to generate a skeleton of subjects with radar signals. This choice to incorporate vision for training RF also comes as recent works in the vision field have increasingly produced better models for handling the limited resolution of mmWave radar. Again, these works require subjects to be facing the camera while performing their activities. Additionally, such approaches can also require large amounts of labeling, increasing the training overhead, while also leading to privacy implications. Regarding repetition counting, one work utilizing a wireless modality attempts to address the problem of repetition counting with WiFi beamforming reports delivers a modest performance at an average absolute counting error of 1.73 [24] owing to its lack of information on the motion of users that WiFi cannot provide due to its low bandwidth. Further, none of these RF-based works have addressed the joint repetition counting and classification problem to aid with cataloging.

Thus, existing works in RF point cloud or doppler-frame profiles [30] have not addressed robustness by learning view-agnostic features, while those that do in CV focus on snapshot 2D images, making it hard to leverage for 3D unstructured data. Multi-view pose agnostic sensing systems using a combination of IMU, acoustic, and cameras have been demostrated before, but for all of these sensors will then be required during deployment [23] thus increasing the deployment overhead. Hence, in designing a practical automated HAC system, we seek to bridge this gap in learning view agnostic features directly from the more informative yet unstructured RF point clouds with minimal labeling, and without assistance of additional modes of sensing, like that of vision or IMU, or multiple devices in deployment. Additionally, we address the largely unexplored, yet highly challenging problem of repetition counting on 3D point cloud sequences to segment point cloud videos and isolate different instances of exercises, which forms an integral component of any HAC system, especially exercises.



(a) (b)

Fig. 2. (a) 10 exercises (b) Training and deployment set up

3 NEED FOR DIVERSITY

3.1 Experimental Study

The objective of this work is to build an automated, easily deployable HAC system for exercises using RF alone. We consider a comprehensive dictionary of 9 exercises, some more challenging than the others, while spanning both the upper and lower body, as shown in Fig. 2a.

To understand the challenges facing existing solutions, we conduct a simple experimental study, where human targets are made to perform various exercises in front of a single radar or 3 radars (provide 3 different perspectives) as shown in Fig. 2b. Each commodity radar has a 12x16 aperture (details in Section 5). We employ a state-of-the-art ML-model (P4 [18]) capable of learning 4D spatio-temporal features from 3D point clouds and train it with data from a single radar or from all 3 radars. The results are shown in Figs. 4a and Fig. 4b, categorized either based on the target's facing direction (with respect to the reference radar) or the exercises. Four important observations can be made:

(i) Compared to 2D range-doppler spectrums often employed in prior art, leveraging 3D point clouds offers the most distinguishable information (conducive for ML models) for radars to classify different exercises as seen in Fig. 3. As shown in Fig. 3, the primary doppler component are found in two different locations (different X values), yet on the range doppler spectrum, the negative velocity peak appears at roughly the same range and velocity.

(ii) When trained with a single radar data (Fig. 4a), the exercise classification accuracy is only around 70% even when the target faces the radar (0 degrees). Unlike cameras, the sparse reflections from mmWave radars, makes it highly challenging for the model to differentiate between different exercises, which are a complex sequence of multiple gestures. It is easy for the model to get confused between exercises, whose limited point cloud data might appear similar, especially when the exercise dictionary is large.

(iii) The accuracy quickly drops to 50% once the target faces away from the radar, highlighting the fragility of the models to orientation. Orienting away from the radar can compound the model's confusion – one exercise's side view could appear similar to the front view of another exercise. For example, in Fig. 4b, squats in front-facing (sq P4-1R) yields nearly 60% error due to the model miss-classifying it as lunges (ln P4-1R). Also, the errors are as high as 80% for lunges and push-ups as they look very different when not facing the front view.

(iv) Adding multiple perspectives of the same exercise (through radar diversity) helps the model learn discriminating features across exercises better as we see in Fig. 4a. While the accuracy improves from 70% to 90% when the target faces the radar, the errors climb to 30% once the target changes its orientation leaving the model still vulnerable to orientations on which it has not been trained.

Our study highlights the need to address three key technical gaps towards making an automated HAC system practical and robust.

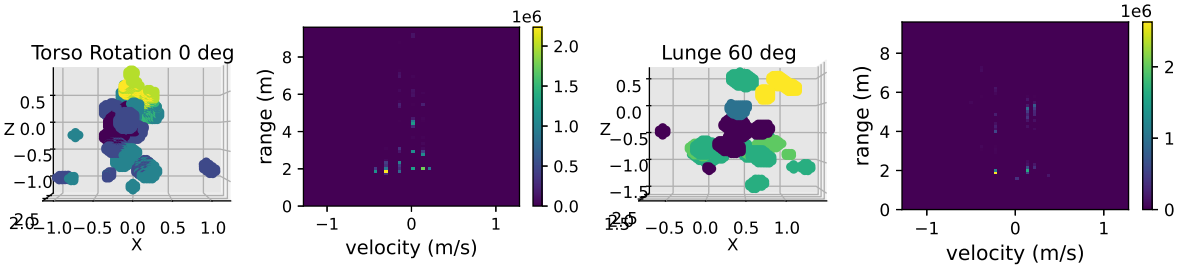


Fig. 3. Difficult to distinguish 2d Range-Doppler spectrums vs corresponding point clouds (Torso rotation vs lunges)

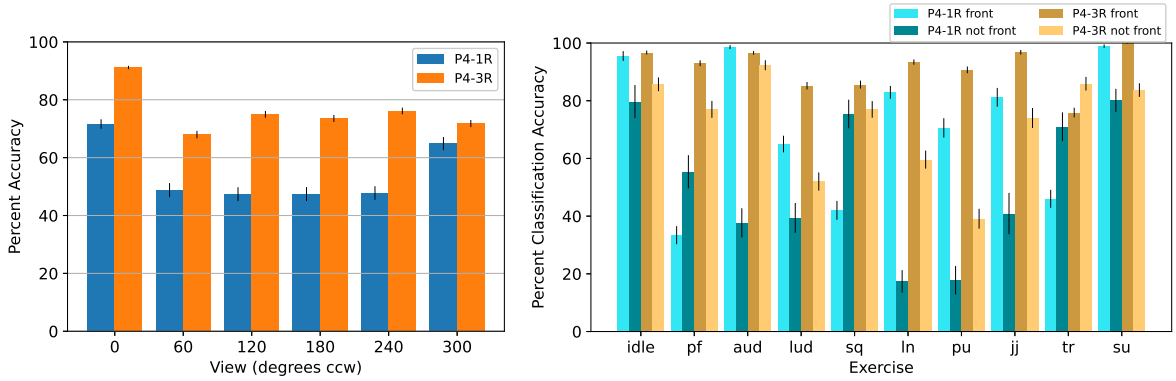


Fig. 4. (a) Motivation (diversity) (b) P4 over exercises

3.2 Challenges

Robustness: Discriminating exercises from a single radar view is highly challenging owing to sparse mmWave reflections and sensitivity to target orientations. While radar diversity (with multiple radars at training) can definitely help the model, it does not provide a comprehensive perspective on all possible target orientations. This highlights the need for the model to leverage diversity and learn discriminative features in an orientation-agnostic manner.

Deployment Overhead: One potential solution is to collect data with targets performing the various exercises numerous times at various different angles/orientations from the radar. Beyond the tediousness of data collection, it also incurs a substantial labeling overhead to account for orientations, not to mention the inability to scale to additional exercises. Hence, we need a solution that imposes minimal deployment burden, not just during operation, but also during training (data collection and labeling) that allows it to easily scale and accommodate new exercises

Automation: Beyond the classification of exercises, HAC systems also need to count the repetitions to deliver utility to their end users. However, repetition counting is a notoriously hard problem even for vision systems, where the state-of-the-art ML-driven approaches [16, 27, 51] yield accuracies of only around 65-70%. Further, they incur a high training and labeling overhead (segmented video streams), adding to the deployment burden. Notwithstanding its accuracy, its application to our mmWave application is further complicated by the sparsity of RF data as well as the need for robustness to various orientations. Fig. 9a exemplifies this difficulty, where the point cloud data corresponding to 3 repetitions of an exercise reveal no explicit patterns of periodicity, unlike vision data.

4 *RF-HAC*: DESIGN

4.1 Overview

Towards addressing the aforementioned challenges, we present *RF-HAC*- one of the first systems to enable practical and robust RF-based human exercise cataloging in real-time. *RF-HAC*'s design incorporates two key building blocks as shown in Fig. 5.

The first component tackles the challenge of exercise classification while not requiring users to be facing the mmWave radar. *RF-HAC* leverages the diversity of multiple radars along with contrastive learning to create a novel view-agnostic self-supervised transformer-based model that will be deployed on a single radar. Three radars (separated by 120^o around sensing region of interest) are employed during training to simultaneously produce point clouds from different views – these are not only used to create positive (e.g. same exercise, different views) and negative samples (e.g. different exercises) easily for contrastive learning, but also aggregated into a holistic, dense 3D point cloud. Rotating the latter allows *RF-HAC* to synthetically generate numerous views of the target (covering 360 degrees), providing additional contrastive samples from various orientations, all while requiring users to perform the exercise in only one orientation, thereby eliminating the tedious view-specific deployment overhead associated with other approaches. This is complemented by *RF-HAC*'s model that is trained by pitting the different radar views against each other through a mutual information based loss function that balances accuracy with robustness across views, allowing it to learn only the exercise-discriminating features from the 3D RF point clouds without carrying any artifacts specific to the orientations.

The second component tackles the challenge of counting the number of repetitions of an exercise that a user/target performs. Here, *RF-HAC* leverages a valuable pattern in the doppler values of point clouds generated during an exercise, namely alternating positive and negative doppler values that correspond to the two halves of an exercise repetition (e.g. sitting down and standing up in a sit-up exercise). *RF-HAC* starts by exploiting the high correlation between proximal (spatially) points in a target's point cloud, to interpolate the doppler values of the unstructured 3D point cloud and map them to a structured grid-based 3D point cloud. The latter's self-similarity features are then analyzed across decomposed 2D planes (for reduced complexity) to identify any alternating regions of positive and negative similarity (as shown in Fig. 8), and hence determine the appropriate segmentation of the data stream, leading to its repetition count. Leveraging the underlying doppler behaviour allows *RF-HAC* to adopt a non-ML based robust segmentation approach, which not only eliminates any deployment overhead but also avoids sensitivity to target orientations and faciliates real-time operation.

Finally, RF-HAC's segmentation and classification modules synergistically reinforce each other during operation. The incoming sensor stream is segmented and fed to the classification model, which helps further boost the latter's accuracy by eliminating multi-count and partial exercise segments. On the other hand, the model's identification of the exercise (after several frames) is used to condition the segmentation algorithm, increasing its counting accuracy for challenging exercises (e.g. involve alternating limbs), whose features reveal multiple harmonics.

4.2 Leveraging Radar Diversity

RF-HAC leverages the diversity gain from multiple radars during training. It deploys three radars, each being angularly offset by 120 degrees from each other with d_n being its distance to the center of the sensing region from radar n as shown in Fig. 2b.

Each point within a radar's point cloud has five values (x, y, z, doppler, and intensity) recorded with it. However, the x, y, and z values are from a given radar's point of view. Hence, to aggregate point clouds, *RF-HAC* applies offsets to radar data to bring the points to appropriate locations within the common frame of reference of a single

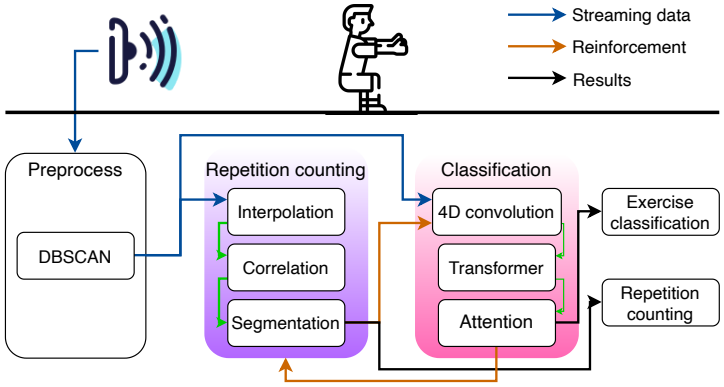


Fig. 5. Deployed system setup

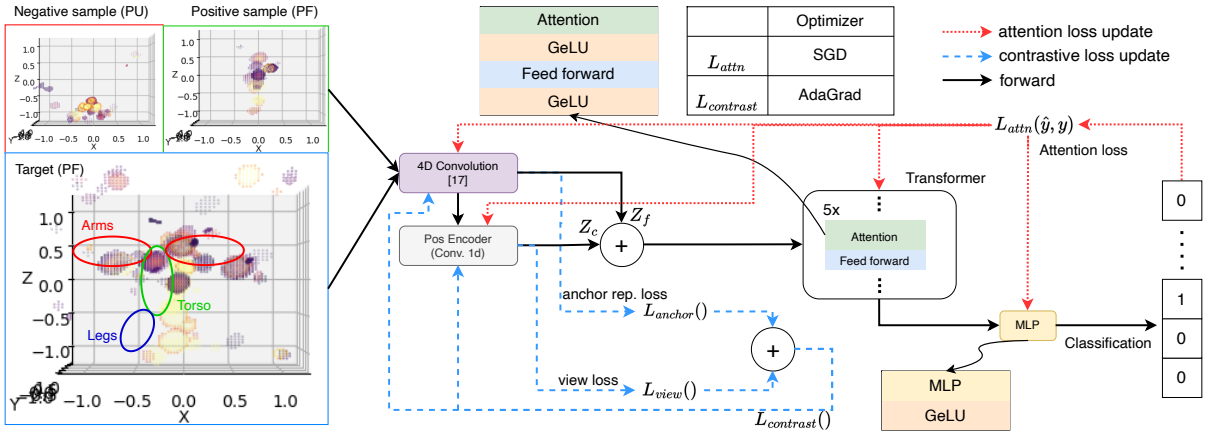


Fig. 6. View-agnostic model

radar. Let $\delta x, \delta y$ be the translational offset of one radar n with respect to a reference radar (say 1) and $\delta\theta$ be its corresponding relative counter-clockwise azimuth offset around the center point as shown in Fig. 2b. We have,

$$\delta x = \quad d_n \sin(\delta\theta) \tag{1}$$

$$\delta y = \quad d_1 - d_n \cos(\delta\theta) \tag{2}$$

Using δx and δy, the new position (x', y') of a point given its original position (x, y) is calculated as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\delta\theta) & \sin(\delta\theta) \\ -\sin(\delta\theta) & \cos(\delta\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} \tag{3}$$

By aggregating several uniformly spaced (angularly) point clouds into a 360 degree RF point cloud, *RF-HAC* avails two benefits: (i) any clutter (noise and interference) are dispersed, making it easier to filter them through a clustering algorithm such as DBSCAN [17]; (ii) a point cloud representing a 360 degree field of view of a target can be flexibly rotated. Rotating a point cloud from a single radar is less a view of a different radar and more just a rotated version of the radar's field of view. Thus, the diversity benefit of having multiple radars allows *RF-HAC* to generate a lot of orientation samples for a given exercise without encumbering the target to perform the exercise in more than a single orientation, thereby keeping the overhead minimal. This also facilitates its robust classification as we shall see shortly.

4.3 View-agnostic Activity Recognition

A key benefit of *RF-HAC* is its view-agnostic recognition feature, making it easy to deploy and use – users only need to stand within the radar's sensing range regardless of their orientation and position. To achieve this, *RF-HAC* designs a contrastive pose recognition model to use alongside a multi-headed attention transformer-based exercise classification model.

4.3.1 Model. At a high level, to arrive at the classification output, the point cloud sequence first goes through 4D convolution [18] and then a transformer. As the name suggests, 4D convolution learns the structure of a 3D point cloud over time (four dimensions) by capturing the displacements of the points for a given kernel as features, and fusing them with features corresponding to their doppler and intensity values. Into the 4D convolution, we input a given point cloud sequence along with its input features, which in *RF-HAC* is the doppler and intensity values associated with each point. To capture the structure of the point cloud, 4D convolution calculates a set of anchor points, which become Z_c, and each point's displacement from those anchor points. The displacements are featurally fused with the input features, doppler and intensity, to obtain the output features, Z_f. These two outputs of 4D convolution (Z_c, Z_f) are fused into one set of features, then fed into a Transformer module made up of multiple multi-headed attention layers. With temporal features being largely learnt in the Transformer module, the latter is integral to classification of exercises over a period of time longer than one frame.

4.3.2 Loss Functions. RF-HAC introduces and employs three different loss functions, each of which is described below.

The first is the attention loss. The attention loss is not a loss function for the self-supervised contrastive learning, but rather the loss function for the classifier. The attention loss, $L_{attn}(\hat{y}, y)$ is captured as a cross entropy loss of the classification weights and a single target class denoting which exercise the clip is depicting. Mathematically, $L_{attn}(\hat{y}, y)$ can be written as:

$$L_{attn}(\hat{y}, y) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}) \tag{4}$$

The classifier utilizes view-agnostic features generated from the second and third loss functions, related to contrastive learning, to classify more accurately. These loss functions are view loss and anchor representation loss respectively. The view loss estimates the mutual information between given features (Z_f) and a set of positive (Z_f^+) and negative (Z_f^-) features. Since these positive/negative features are derived from a point cloud video with a different view/exercise (positive/negative samples), this would help the classification model learn features that are common (specific) to an exercise even when its point clouds correspond to a different view. The view loss (L_{view}) is computed using estimated mutual information between a given point cloud and its positive and negative samples [52]:

$$L_{view}(Z_f, Z_f^+, Z_f^-) = \mathbb{E}[\rho(-f(Z_f, Z_f^+) \odot I^+)] - \mathbb{E}[-\rho(f(Z_f, Z_f^-) \odot I^-)]$$

where $\rho(x) = \log(1 + e^x)$ is the softplus activation and f computes the similarity between features (using an approximated Jensen-Shannon mutual information estimator [29]). I^+ and I^- are positive and negative indicator matrices computed by comparing the labels of different samples within a batch. The anchor representation loss (L_{anchor}), meanwhile, helps to fix the features from a different orientation (at the user) and features from a different view (at the radar) to the same global frame of reference. This means that from the model's perspective, the user changing their orientation by 60 degrees counter-clockwise, for example, should have the exact same effect as the radar getting rotated around the user by 60 degrees clockwise. It is computed as:

$$L_{anchor}(Z_c, Z_f) = \mathbb{E}[\rho(-f(Z_c, Z_f) \odot I^+)] - \mathbb{E}[\rho(-f(Z_c, Z_f) \odot I^-)]$$

where, the similarity between the fused features and the input (approximated as Z_c) is captured to ensure the learnt features are appropriately anchored to the input. Finally,

$$L_{contrast} = \alpha L_{view} + (1 - \alpha) L_{anchor} \tag{5}$$

where α is adjusted to balance the magnitude of the two contrastive loss functions.

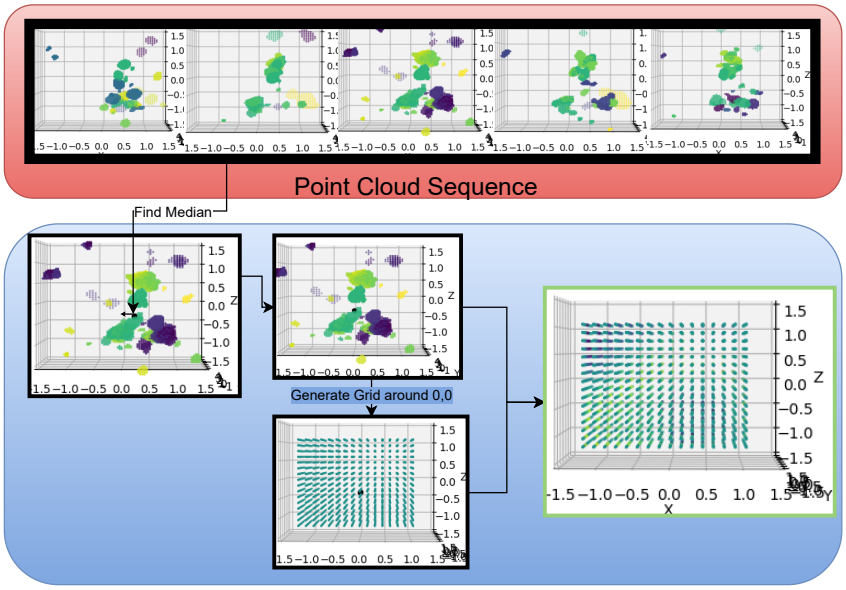


Fig. 7. Interpolation operation

4.3.3 Pipeline. The overall flow of the model is shown in Fig. 6. After sampling positive/negative clips for a given batch, the model takes all of the clips (local, positive and negative) and applies it to a single 4D convolution layer[18]. Then, the model would utilize the representative vectors of positive and negative feature data (Z_f^-, Z_f^+) derived within the batch along with the target's representative vectors Z_f, Z_c and apply Equation 5 to calculate the contrastive loss. Then, the model would embed Z_f, Z_c into a single feature vector before being fed into several transformer layers and an MLP (multi-layer perceptron) layer to get the final exercise prediction $\hat{y}$. Finally, the model would evaluate $\hat{y}$ with ground truth exercise y to obtain attention loss, indicated by Equation 4 and aggregate the overall loss to train the model, where the overall loss is the following:

$$L_{overall} = L_{attn} + \lambda L_{contrast} \tag{6}$$

where λ is applied to balance the attention and contrastive losses, and consequently the model's accuracy and robustness (to views). We discuss the optimal values of α and λ in Section 6.1.4.

The novelty of *RF-HAC*'s model is to intelligently bring together temporal learning capabilities enabled by 4D convolution and transformers, and pose-agnostic learning enabled by contrastive approaches on 2D images, to enable pose-agnostic temporal learning for 3D RF point clouds.

4.4 Robust Activity Counting

Counting the number of repetitions of an exercise accurately (along with its duration) is integral to any exercise cataloging, and is vital to post-analysis. Also, while the model accepts a stream of frames in a clip to classify the exercise, giving it some robustness to varying exercise speeds, vastly differing speeds however can potentially impact its accuracy. Hence, in addition to tracking repetition count, an accurate counting algorithm can lead to better clip segmentation and hence further improve the model's accuracy.

Some recent works have explored ways to count repeating signals, including a system that tracks heart rate with mmWave radar by correlating with a template that is learned via deep learning [21]. However, most of these algorithms are designed to work on 1D signals or structured 2D images [16] over time. This leaves repetition counting of unstructured 3D point clouds as a mostly unexplored problem.

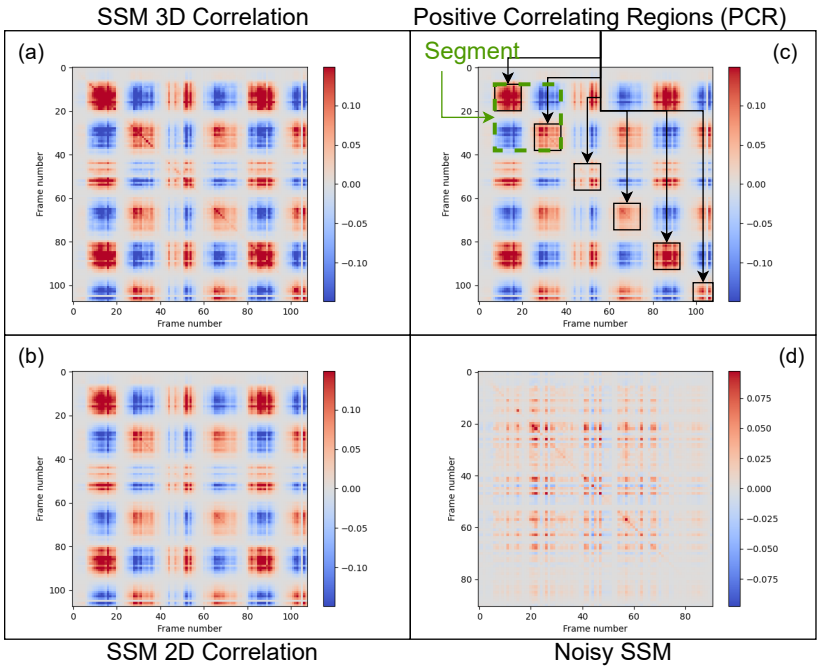


Fig. 8. Assorted SSMs

4.4.1 Un-structured to Structured Point Clouds. RF point cloud videos contain a sequence of unstructured 3D point clouds that make pattern analysis difficult. Hence, *RF-HAC* first maps the incoming unstructured 3D point cloud into a more structured (grid-based) 3D point cloud.

To obtain a structured 3D grid from the unstructured 3D point cloud, we first must determine where to place this 3D grid and how large this grid should be. Since *RF-HAC* focuses mainly on exercise tracking for indoor environments, we can expect users to approximately stay in the same place when performing an exercise. Thus, the 3D grid is placed around a median point, computed as the median of all x values and all y values, with z values remaining unchanged as shown in Fig. 7. The points on this grid are weighted using doppler values, where the doppler value of a grid point is obtained by interpolation of doppler values of points in it's neighborhood. This leverages the underlying high correlation between doppler values of spatially proximal points that in turn correspond to a part of the user's body. *RF-HAC* employs Delauney Triangulation [26] for interpolation and is illustraed in Fig. 7. The result of this interpolation is a matrix $\tilde{V} \in \mathbb{R}^{n \times d \times d \times d}$, where each element in $\tilde{V}$ is a doppler value of a point in the grid. If n is the number of frames, d is the size of the dimensions of the grid, and p is the number of points per frame, then the runtime complexity incurred by the interpolation step is $O(np \log p) + O(nd^3)$.

4.4.2 Self-similarity matrix (SSM). After obtaining a grided point cloud $\tilde{V}$ weighted with doppler values, we spatially correlate each frame to form a self-similarity matrix (SSM). However, correlating directly in 3D space can be very computationally expensive with a complexity of $O(n^2d^3)$. With n and d being comparable in size, the processing of 3D SSM ($O(d^5)$) could end up becoming more expensive than the interpolation step ($O(d^4)$). In contrast, *RF-HAC* decomposes the 3D correlation problem into three 2D correlation problems, by mean pooling a given dimension, and correlating the resulting 2D plane, and subsequently summing the result for of all the three planes. This process has a computational complexity of $O(nd^3) + O(n^2d^2)$, where the savings of $O(d)$ proves valuable in *RF-HAC*'s real-time capability. From Fig. 8a and Fig. 8b, it's evident that the SSM produced by first

mean pooling each dimension and then summing up the correlations in 2D and the SSM produced by directly correlating in 3D are very similar. Thus, if we have interpolated grids $G^{(i)}$ and $G^{(j)}$ where i, j are frame numbers, we obtain the doppler correlation between the two frames (D_{ij}) as:

$$D_{ij} = \mathbb{E}[\mathbb{E}[G_{xy}^{(i)}]\mathbb{E}[G_{xy}^{(j)}]] + \mathbb{E}[\mathbb{E}[G_{xz}^{(i)}]\mathbb{E}[G_{xz}^{(j)}]] + \mathbb{E}[\mathbb{E}[G_{yz}^{(i)}]\mathbb{E}[G_{yz}^{(j)}]]$$

where $G_{xy}^{(i)}$ are the doppler values at frame i at grid points x and y, which in this case, corresponds to a vertical line of points in the z direction.

4.4.3 Leveraging doppler behavior inherent to exercises. From Fig. 8c, it is apparent that the output of the SSM shows alternating regions of positive and negative correlation. This is because when an exercise repetition (say, a sit-up) is performed, it involves two parts: an *exerting* motion is performed at first (e.g sitting down), followed by a *recovery* motion (e.g. standing up) in roughly the same area of space. Thus, when the radar tracks this movement, a positive doppler is detected initially and then when the subject prepares to move back to the original position to then repeat the exercise again, a negative doppler would appear in a similar region of space. When a frame with mostly positive doppler is correlated with a frame with mostly negative doppler in a similar region of space, we obtain a negative correlation as seen in Fig. 8c. Thus, *RF-HAC* leverages this inherent doppler behavior in exercises to detect a "count" by searching for an alternating pattern of positive and negative similarity, namely a 2x2 correlation "checkboard" pattern as seen in Fig. 8c. Note that, with the exerting and recovery motions following each other in a repetition, we are interested in detecting such correlation between frames that are close to each other, i.e. the 2x2 checkboards that touch the main diagonal (diagonal points are non-negative, since $D_{ii} \geq 0$). However, as with any RF system, noise can affect both the points and the doppler values as shown in Fig. 8d. To this end, *RF-HAC* designs an algorithm that can effectively manage these noisy regions.

Algorithm 1 Segmentation given the SSM

Require: Self-similarity matrix D
1: $i_n \leftarrow$ indices of diag(D)
2: $P \leftarrow$ sort i_n based on value of $D_{i_n i_n}$ in descending order
3: **for** p_i in P **do**
4: Compare p_i with p_j, p_k such that $p_j < p_i < p_k$ and $j, k < i$
5: Depending on the values of p_j, p_k, include p_i in a region or form a new region
6: **end for**
7: $R \leftarrow$ all of the formed regions in chronological order
8: **for** All possible R_i, R_j pairs **do**
9: $D'_{R_i R_j} \leftarrow \sum_{m \in R_i, n \in R_j} D'_{mn}$
10: **end for**
11: $S_{R_i R_j} \leftarrow D'_{R_i R_i} + D'_{R_j R_j} - 2D'_{R_i R_j}$ for set S and $i < j$
12: **return** elements in $S_{R_i R_j}$ that give the best possible score without overlap

4.4.4 Segmentation algorithm. We use Fig. 9b to exemplify segmentation in action for sit-ups (SU) while Algorithm 1 shows the steps of the segmentation algorithm in pseudocode. *RF-HAC* starts with frames having large self-correlation. Let $\{i_n\}$ be the sequence containing the indices of these frames from largest to smallest self-correlation. However, the self-correlation alone is not sufficient to tell us which group of high self-correlation indices correspond to exerting motion and which correspond to recovery motion. Hence, *RF-HAC* looks for negative similarity regions, based on which, it splits the frames into different groups called "positive correlating regions" (PCRs), wherein each frame in the group has a positive similarity with other frames as seen in Fig. 9b.

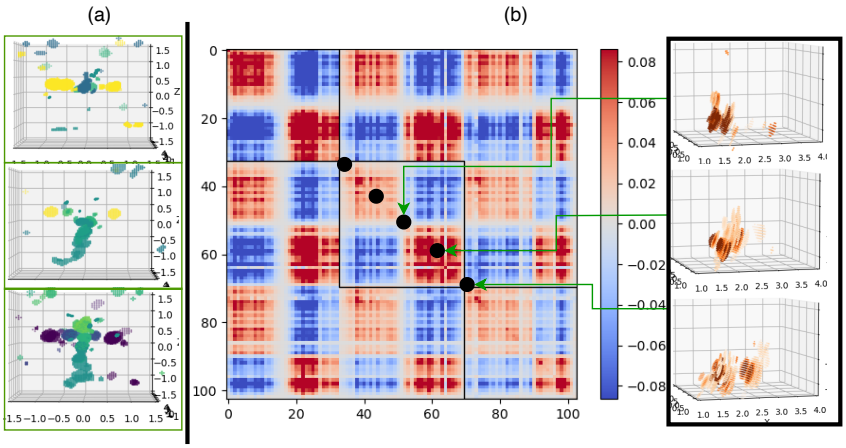


Fig. 9. (a) No explicit patterns in 3 reps (PF) (b) Example segmentation output (SU)

Note that positive and negative correlation regions in SSM, namely the PCRs, are across the entire set of frames, while the frames within each of these PCRs will exhibit positive correlation locally.

Step 1 (Form PCRs): To perform this split as shown in lines 2-6 in Algorithm 1 , *RF-HAC* takes the sequence $\{i_n\}$ and iterates through the frames. At frame index i_j, all frames i_k with index $k < j$ can be called "seen" frames (already processed). When considering whether i_j should be placed in an existing positive correlating region or form a new positive correlating region, *RF-HAC* checks the similarity in D between index i_j and neighboring "seen" frames. This process repeats until every frame becomes seen.

Step 2 (Recursive Search): Once these PCRs are formed, *RF-HAC* subsequently picks and combines these regions to segment the video as seen in lines 7-11 of Algorithm 1 . Assume $\mathfrak{R} = \{R_1, R_2, ..., R_N\}$ is the set of all N regions for the given video. Since we are looking for 2x2 checkboard patterns on the main diagonal, between two segmentation boundaries, the first PCR in the sequence of picked regions, R_{E_1}, as an exerting (E) region, and the second PCR in the sequence R_{C_1} as a recovery (C) region. Let $R_m \rightarrow R_n$ operator denote that chronologically, region R_n is encountered after region R_m. To establish segmentation boundaries, we want to best match an E region with an C region, while conforming to the following constraints:

- $R_{E_1} \rightarrow R_{C_1} \rightarrow R_{E_2} \rightarrow ...$ is alternating for a feasible combination $E_1, C_1, E_2, ...$
- $\{R_{E_1}, R_{C_1}, R_{E_2}, ...\} \subseteq \mathfrak{R}$, indicating that not all regions are necessarily an "exerting" or "recovery" region.

To find the optimal segmentation, *RF-HAC* recursively searches every possible sequence of E and C regions and scores each possible assignment. Let R_n be the n-th region formed in chronological order, then R_n is a set containing the indices of all the frames within the corresponding PCR. Let $D_{R_m R_n}$ be the sum of all correlations between frames of regions R_m and R_n. Mathematically, this can be written as:

$$D_{R_m R_n} = \sum_{j \in R_m} \sum_{k \in R_n} D_{jk} \tag{7}$$

Step 3 (Maximize Score): Then in line 12 of Algorithm 1 , to obtain the best possible segmentation with N_S segments, *RF-HAC* maximizes:

$$\max_{E_1,E_2,...,E_{N_S},C_1,C_2,...,C_{N_S}} \sum_{i=1}^{N_S} D_{R_{E_i} R_{E_i}} + D_{R_{C_i} R_{C_i}} - 2D_{R_{E_i} R_{C_i}} \tag{8}$$
$$\text{s.t. } 0 \leq E_1 < C_1 < E_2 < ... < C_{N_S}$$

where E_i and C_i are the chosen exerting and recovery regions for segment i, with the score function designed to leverage the negative correlation between R_{E_i} and R_{C_i}. Solving this optimization, *RF-HAC* obtains both the optimal number of segments N_s^* as well as its corresponding segments to obtain the repetition count for the exercise (along with the exercise's duration). With a brute-force approach to solving the optimization incurring $O(2^n)$ time complexity due to it's recursive nature, a dynamic programming approach is adopted in *RF-HAC*. Although the latter's worst case complexity could be $O(n^4)$, it occurs only if each frame forms its own PCR, whose probability is negligible owing to the nature of exercises. In the average case, it runs in $O(n^2)$ in practice, making it amenable for real-time implementation.

4.5 Integrating Segmentation-Classification

While each of *RF-HAC*'s classification and segmentation can operate well on their own, their performance can be further enhanced through reinforcement. While the model can fairly accurately classify exercises at a 24-frame (1.5 secs) clip-by-clip level, it can occasionally falter for certain exercises, owing to not knowing when one exercise repetition actually ends and another begins. Additionally, pooling clips within a segment together to make a classification can improve the accuracy of the model. The segmentation algorithm, meanwhile, can tell us when an exercise begins and ends, but can be confounded by certain exercises (especially involving limbs) that can generate multiple harmonics of exerting and recovery motions within a repetition (e.g. alternate leg rises). Each of the exercises exhibiting multiple harmonics have a unique pattern in which it exhibits multiple harmonics even when the difference in user's orientation is considered. For example, arms up-and-down (AUD) always exhibits the exerting and recovery regions twice per repetition. Therefore, by providing exercise classification information from an initial sliding window classifier to the segmentation algorithm, the segmentation algorithm can use this exercise information to accordingly fuse segments produced by the output of the segmentation algorithm for a more accurate segmentation. *RF-HAC* integrates these two components for real-time operation as shown in Fig. 5. The classifier's robustness is improved by pooling the clips within the segment and eliminating multi-count and partial exercise segments. Meanwhile, the segmentation is improved by the classifier providing information on the exercise (available after a few frames), which helps the segmentation address multiple harmonic scenarios by adjust the segmentations accordingly.

5 *RF-HAC*: IMPLEMENTATION

Hardware and software. We implement *RF-HAC* with a commodity 77 GHz mmWave radar [6] that consists of three TI collocated 3x4 radar chips to provide a 12x16 virtual MIMO radar with sufficient antenna gain at 77 GHz. The radar collects real-time mmWave signals, processes them on-board to translate them into a fairly dense (1-4 thousand points per frame) 3D point cloud data (each point capturing 3D coordinate, doppler, and intensity), and then sends them through WiFi 802.11n at the rate of 16 frames/sec to a laptop. The laptop is equipped with Intel Core I7 with 16 cores and 16 GB memory. It is responsible for controlling the 3 radars to simultaneously collect point clouds (Fig. 2b), aggregate them into a 360-degree view and store it for subsequent training, as well as real-time deployment of the trained model on one of the radars. The overall system setup is shown in Fig. 10a. We also use an Intel RealSense D455 depth camera to collect ground truth video for reference alone. It is worthwhile to note that *RF-HAC* does not need the video data either for training its model or for deployment.

Data Collection: To evaluate *RF-HAC*'s generality, we collect data from several environments with a total of 12 different subjects performing numerous exercises in 6 different environments. A total of 1,500 exercise point cloud streams (called videos) were collected that span a total of about 226,000 frames in about 191,000 clips, where clips are generated by a sliding window across frames. Leveraging *RF-HAC*'s ability to learn pose-agnostic features, only the orientations corresponding to the 3 radars (0^o,120^o,240^o) are employed for training (collected simultaneously when subject faces just one radar), which correspond to 550 videos. All the remaining videos that span all possible orientations, are used to test *RF-HAC*'s robustness to unseen orientations.

Operation	CPU Time	Memory	GPU Time	GPU Memory
Interpolation	0.2529 s	943.12 MB	-	-
Correlation	0.7955 s		-	-
Segmentation	0.0756 s		-	-
Classification	-	-	0.0143 s	2705 MB
	Overall process time			
Total time	1.1383 s			

Table 1. *RF-HAC* system profiling (second/clip)

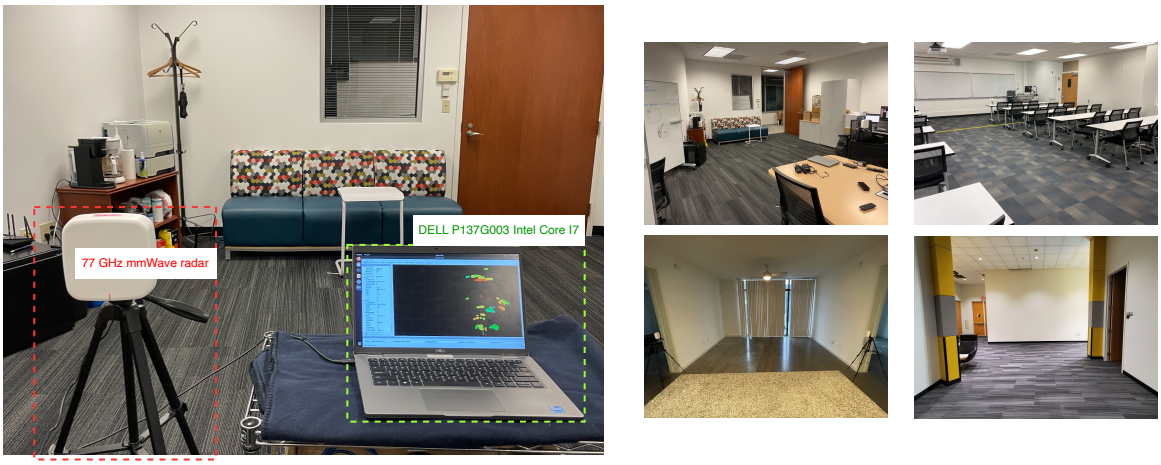


Fig. 10. (a) System setup (b-e) Four of six environments

(i) Environments: To represent everyday settings with varied characteristics, we consider 6 different environments with varied amount of clutter. These environments consists of labs, classrooms, apartments/homes, and hallways, some of which are shown in Figure 10b-e.

(ii) Exercises: We consider a comprehensive set of 10 exercises (including idle) as shown in Fig. 2a. The exercises are diverse enough to cover both upper and lower body, and involve both standing as well as sitting on the ground to understand *RF-HAC*'s classification capability. They are also chosen to cover a range of speeds (paces), varying from static (idle) to moderate (e.g. pectoral fly) to highly active (e.g. jumping jacks), to understand *RF-HAC*'s counting capability.

(iii) Subjects: The subjects participating in the experiments are of various heights from around 165 cm to 185 cm. Different subjects perform exercises at their own pace, and at different distances from the radar, ranging from 1.5 m meters to 3.5 m, as well as at 5m. In addition to the environments, we also split the 12 subjects, such that 5 subjects' datasets are used for training, while the remaining 7 subjects' datasets are held out for testing.

We provide access to a portion of our dataset and our code [4, 5].

Training: *RF-HAC*'s model architecture is shown in Fig. 6. It is trained with a 4D convolution layer [18] followed by a 1D convolution layer for the point cloud hidden vector and forwarded to the transformer. The transformer is comprised of 5 attention plus feed-forward layers each followed by GeLU [22] activation function. Lastly, there is a multilayer perceptron and GeLU before having the classification result. We use SGD [34] optimizer for attention loss and AdaGrad [15] for contrast loss to update the model. To best utilize the capability of contrastive learning for deployment on a single radar, we not only employ aggregated data in training, but also contrastively learn between point clouds from each of the individual radars.

	Env 1		Env 2		Env 3	
	fr	nfr	fr	nfr	fr	nfr
RadHAR	93.6	46.5	83.0	34.7	67.3	27.1
mTransSee	83.2	50.0	78.9	41.9	83.4	45.0
P4, 1R	88.2	56.6	89.6	43.5	63.9	38.2
P4, 3R	89.8	82.0	90.8	82.6	88.8	90.5
RF-HAC slide window	94.4	93.3	95.3	92.8	93.5	95.8
RF-HAC segments	93.4	92.2	95.3	92.9	91.6	93.5

Table 2. Classification Accuracies in 3 seen environments

5.1 Real-time Capability

We now profile *RF-HAC* to understand its real-time operational capability on off-the-shelf home-owned or edge computing devices. We use Intel(R) Core(TM) i9-10980XE CPU @ 3.00GHz for CPU with 64-bit x86_64 architecture and NVIDIA GeForce RTX 3090 for GPU. However, we use Linux's kernel feature, cgroup [2] to restrict *RF-HAC*'s resource usage so as to emulate the capability of a resource-constrained edge-compute device, similar to Intel NUC [1]. The profiling results from Table. 1 show that the overall processing time for one clip, which spans 24 frames and 1.5s of actual exercise time, takes 1.1383 seconds for the entire pipeline of segmentation and classification. It is clear that *RF-HAC* can process the incoming radar point clouds and catalog the exercises in real-time at a processing rate of 0.75s for every second of exercise, while operating on off-the-shelf edge computing devices that could be co-located with the radar (or integrated into a single device in the future). This latency can be further reduced by reducing the number of grid points in the interpolation step at a potential cost of segmentation accuracy. Note that, while *RF-HAC* currently requires a GPU for exercise classification, which are available on several recent edge computing devices [1], it is also possible for its classification model (especially 4D convolution) to be optimized for operation without a GPU in the future.

6 EVALUATION

Baselines: In evaluating *RF-HAC*, we also consider a version where the incoming data stream is not segmented (RF-HAC-ns) and evaluated using purely a sliding window, and compare it to two relevant baselines: one from vision – a state-of-the-art point cloud based pose recognizer (P4) that is trained on both single radar (P4-1R) and aggregated 3 radar (P4-3R) data, and another recent RF work (RadHAR) [37] that employs mmWave point clouds for activity recognition. For RadHAR, we adjust the sliding window size to 32 frames to match its 2 second windows, as well as decimate our point cloud as needed to match the radar density used in RadHAR. We also implement another RF baseline (mTransSee) [30] that performs classification in an environment-independent manner using 2D doppler-frame profiles as input to their model. We curate these doppler-frame profiles by summing up intensities in a doppler bin as specified by mTransSee.

For repetition counting, we anecdotally show a result from a different modality. Namely, a WiFi beamforming reports based repetition counting scheme called CBR-ACE [24].

6.1 Robustness in Classification

6.1.1 Overall performance. Table 2 presents the overall exercise classification accuracy for three environments. Three inferences can be made: (i) RadHAR, mTransSee, and P4-1R each lacks robustness and suffers considerably in non-front facing orientations (as low as 27%, 42% and 38% respectively in Env 3) of the subjects compared to the front-facing scenarios. Despite RadHAR's ability to classify front-facing exercises with reasonable accuracy, the sparse point clouds and dependence on subject orientation inherently limits its ability to accurately classify non-front facing views. Meanwhile, mTransSee's loss of information from abstracting away the position of the points causes it to not be robust. Additionally, the doppler profile can change quite significantly when the user does not face the radar resulting in the drop in classification accuracies for non-front facing scenarios. (ii) Adding radar diversity in data collection allows P4-3R to appreciably bring down this performance gap. However, while

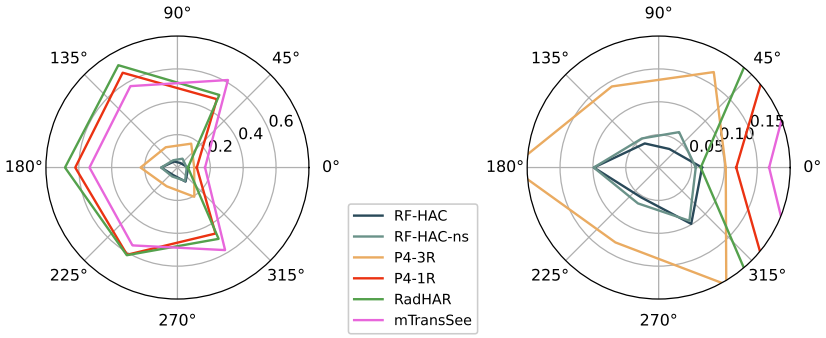


Fig. 11. Classification error vs view

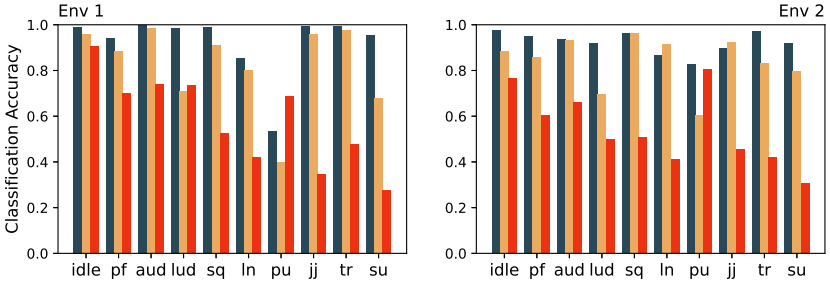


Fig. 12. Classification accuracy vs exercise

the smaller validation set in Env 3 contributes to a decent accuracy of 90% (evident in Table 3 since most of the exercises classified with "100% precision"), the much larger validation sets found in Env 1 and 2 shows that there is still a 8% performance gap for P4-3R between front and not-front facing (iii) *RF-HAC* delivers a robust performance at over 90% accuracy on all of the environments, regardless of front-facing or not front-facing. While training on the seen environments allows the sliding window classifier to better fit the dataset and perform slightly better than its segmented counterpart, we will see the robustness benefits of the segmented classifier later for unseen environments.

6.1.2 Performance by view. We now dissect performance to understand the impact of subject's orientation angle in Figure 11. We observe the following: (i) *RF-HAC* delivers over 90% accuracy for all orientations. (ii) P4-3R is able to deliver close to 90% accuracy for front facing (0^o), but drops to 86% for (120^o, 240^o), directions for which it was still trained on, and further drops to 83% for 60^o under 80% for 180^o and 300^o. Thus, while radar diversity in data collection helps, it is not sufficient if the model is incapable of extracting view-agnostic features, unlike *RF-HAC* (iii) P4-1R, mTransSee, and RadHAR performs at a decent 88%, 83% and 93% respectively for front facing alone but suffers significantly in other directions, owing to lack of both data diversity and robustness. mTransSee in particular suffers in the 60^o and 300^o directions as the doppler profiles, when viewed as images, are a contraction of the original. Therefore, these contracted doppler profiles can easily be confused with one another.

6.1.3 Performance by exercise. Analyzing the performance based on individual exercises in Fig. 12 helps us understand where the models face challenges. Several interesting observations are in order: (i) *RF-HAC* delivers a maximum gain of 25-30% over P4-3R for four of the exercises, namely lunges, push-ups, sit-ups and legs up-down, all which involve a significant portion of the exercise closer to the ground. Reflections collected from the legs' height or lower are often sparse and present the most challenging scenarios for feature extraction with existing models, compared to upper body exercises. (ii) Push-ups is by far the most challenging exercise for the models. While *RF-HAC* delivers over 90% accuracy for almost all exercises, accuracy on push-ups suffers in the two heavily

Exercise		IDLE	PF	AUD	LUD	SQ	LN	PU	JJ	TR	SU
Env. 1	Precision	98.9	94.1	100.0	98.6	98.9	85.5	53.3	99.5	99.6	95.4
	Recall	98.4	100.0	96.1	99.1	83.4	99.0	92.3	99.1	98.5	66.4
Env. 2	Precision	97.2	93.9	97.6	94.6	96.2	89.5	79.4	96.0	96.9	97.8
	Recall	97.3	96.2	97.6	96.8	84.5	97.7	97.3	96.0	96.2	82.4
Env. 3	Precision	100.0	95.7	100.0	100.0	100.0	100.0	48.1	100.0	95.6	100.0
	Recall	94.1	100.0	100.0	100.0	90.0	100.0	100.0	100.0	100.0	62.2

Table 3. Precisions and Recalls of *RF-HAC*

Model	Env 1 (nfr)	Env 2 (nfr)	Env 3 (nfr)
No contrastive loss (P4-3R, $\lambda = 0$)	82.0	82.6	90.5
View loss only ($\alpha = 1$)	91.5	90.3	93.9
RF-HAC slide window ($\alpha = 0.1$)	93.1	92.8	96.0

Table 4. Ablation study of different α and λ

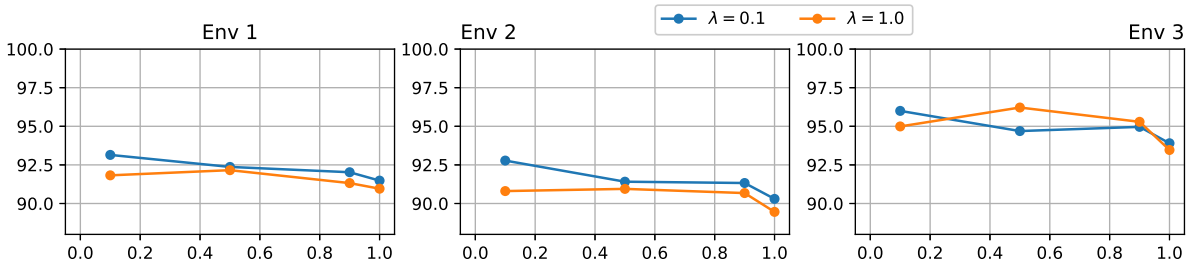


Fig. 13. Ablation study of different α and λ

cluttered environments (Env. 1 and 3), although performs reasonably well in the other environment. This can be attributed to the increased impact of multipath in the first environment that obscures the point clouds (even after clustering with DBSCAN), and creates confusion between push-ups and sit-ups for the classifier. This can be confirmed by the precision and recall for push-ups and sit-ups respectively in Table 3 for the first environment, where *RF-HAC* performs well in all other exercises.

6.1.4 Ablation study. In choosing the best parameters for *RF-HAC*, we conduct an ablation study with three different values of λ (0, 0.1, and 1) and α (0.1, 0.5, 0.9, 1.0). We note that taking out contrastive learning from *RF-HAC* by setting $\lambda = 0$ gives the equivalent of P4-3R. We underscore the importance of the anchor loss function by removing its loss value from consideration through setting $\alpha = 1$. Based on the study over the three seen environments shown in Table 4 and Fig. 13, we conclude that the values of $\alpha = 0.1$ and $\lambda = 0.1$ yields the best results in our study, which we incorporate into *RF-HAC*. Though Environment 3 shows $\lambda = 1.0$ performing better at some values of α, Environment 3 represents a much smaller dataset compared to Environments 1 and 2, thus this deviation is a result of impreciseness rather than evidence that $\lambda = 1.0$ might be a better choice.

6.2 Segmentation and Counting

Our datasets include point cloud videos capturing both 5, 8, and 10-repetition exercises in various environments. There exist no solutions to automatically segment point cloud videos to generate individual repetitions accurately and reliably. Hence, we focus on evaluating the total repetition count of the exercises. To understand the benefit of exercise classification reinforcing the segmentation algorithm real-time in *RF-HAC*, we also consider a non-reinforced version (*RF-HAC*-ns), that receives non-segmented input. Further, with no existing model to count the repetitions on RF point clouds, we consider a baseline that operate at various fixed rate of segmentation

Exercise		PF	AUD	LUD	SQ	LN	PU	JJ	TR	SU	**Overall (Avg)**	**Overall (Abs)**	**CBR-ACE**
5 reps	Reinforced	0.077	0.081	0.051	0.129	0.051	0.045	0.238	0.121	0.039	**0.092**	0.46	1.73
	Not	0.077	0.85	0.749	0.115	0.051	0.066	0.238	0.121	0.055	**0.258**		
8 reps	Reinforced	0.25	0.094	0.156	0.156	0.0	0.0	0.188	0.031	0.063	**0.104**	0.832	
10 reps	Reinforced	0.075	0.092	0.042	0.1	0.042	0.07	0.133	0.05	0.0	**0.067**	0.67	
	Not	0.075	0.783	0.892	0.083	0.042	0.067	0.133	0.05	0.0	**0.158**		

Table 5. Counting error of segmentation

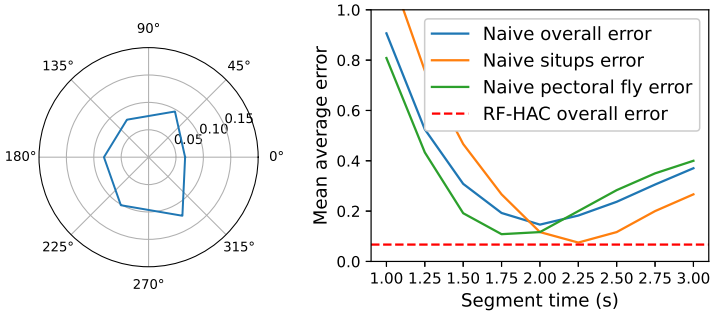


Fig. 14. (a) Counting error (b) Segmentation performance

(corresponding to different speeds of the exercise) and also compare our results to CBR-ACE, a repetition counting algorithm based on WiFi beamforming reports [24].

6.2.1 Overall performance. Results in Table 5 reveal (i) the reinforcement of segmentation with knowledge of the exercise (from classifier after a few frames), helps substantially improve the counting accuracy. This is especially useful for exercises involving limbs such as arms and legs up-down (aud, lud), that result in multiple harmonics owing to their nature. *RF-HAC*'s segmentation can appropriately adapt to these harmonics, resulting in an overall counting accuracy of over 90%. (ii) Jumping jacks (JJ), being an inherently high-speed exercise, makes it challenging for an SSM to capture both the exerting and recovery phases of a repetition for accurate segmentation, yielding the worst accuracy of about 77% for *RF-HAC* with 5 repetitions, i.e. about 1 wrong count. However, with typical exercises spanning more repetitions (10-15), JJ's accuracy improves to 87% with 10 repetitions, while overall accuracy also improves to over 93% (i.e. avg. count off by <1). This substantially outperforms CBR-ACE which produces an average absolute count error of 1.73 [24] as shown in Table 5.

6.2.2 Performance by view. With *RF-HAC* being view-agnostic from a classification standpoint, we also evaluate the robustness of its segmentation algorithm to various orientations of the subject in Fig. 14a. We find only a 6% variation between the highest and lowest accuracy orientations, indicating that *RF-HAC*'s segmentation algorithm, by virtue of leveraging the inherent doppler behavior of exercises, avoids the pitfalls of conventional ML models to deliver robustness to orientations.

6.2.3 Effect of adaptive segmentation. Fig. 14b compares *RF-HAC*'s segmentation with practical baselines that consider various fixed paces (1s, 1.5s, 2s per repetition) for the exercises. With different exercises inherently having different speeds that also vary with subjects, *RF-HAC*'s *automated* segmentation outperforms the best of the baselines, irrespective of the exercise, while also demonstrating the impracticality of the baselines.

6.3 Practical Deployment Scenarios

To understand if *RF-HAC*'s robustness allows for deployment in practical everyday scenarios, we consider performance in completely unseen environments and targets, followed by impact of NLoS and complicated exercise patterns (mixture).

6.3.1 Unseen environments and targets. From Fig. 15, we see that (i) *RF-HAC*'s robustness allows it to deliver an accuracy of 89-94% over 3 different unseen environments; and (ii) *RF-HAC*' classification is reinforced by its

Exercise	PF	AUD	LUD	SQ	LN	PU	JJ	TR	SU	**Overall**
Unseen Env.	0.133	0.048	0.135	0.169	0.017	0.1	0.296	0.077	0.071	**0.116**
Unseen Target	0.133	0.1	0.056	0.044	0.022	0.033	0.178	0.233	0.022	**0.091**
NLoS	0.125	0.35	0.2	0.175	0.225	0.225	0.3	0.15	0.175	**0.214**

Table 6. Mean average counting error for practical scenarios

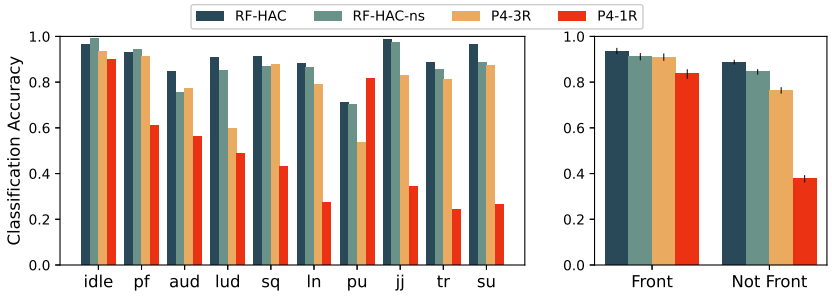


Fig. 15. Unseen environments results vs exercise and view

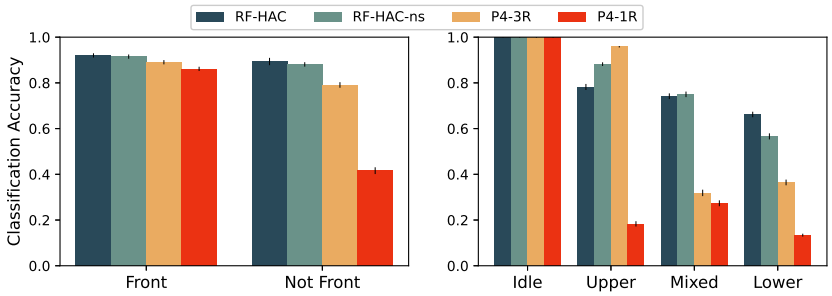


Fig. 16. (a) Unseen targets (b) Partial NLoS

segmentation to improve accuracy (over RF-HAC-ns) by 4-5% for the front-facing and non-front facing scenarios respectively.

RF-HAC's robustness is preserved on unseen (in training) targets as well, as we see from Fig. 16a, where it delivers accuracies similar to those of trained targets in Fig. 12.

6.3.2 Partial occlusion/NLoS. In this experiment, the radar's (placed 107cm high) view to the subject is partially blocked by a 82cm tall couch 1m away in front of the radar with its length perpendicular to the view of the radar. Subjects are then asked to perform exercises behind the couch (from the radar's point of view) facing away from the radar or perpendicular to the radar. Fig. 16b shows that while the accuracy does suffer overall for all the models, *RF-HAC* suffers the least impact, delivering a fair 70-80% accuracy even in this challenging scenario, especially for lower-body exercises (PU, SU, LN, SQ), which P4-3R classifies at around 35% accuracy as a result of classifying them as upper body exercises. Accuracies for P4-1R are close to random guessing. The segmentation algorithm is impacted appreciably owing to a large number of points missing in the point cloud, but is still able to segment at around 20% mean average error (Table 6). Though the results are substantial, some limitations of this study include that the couch was not very reflective, thus getting filtered out by DBSCAN, and the fact that some parts of the knees were still visible to the radar, thus enabling classification despite the missing data. Therefore innovations in intelligent surfaces could help further increase robustness to such NLoS deployments.

6.3.3 Exercise mixture. In practice, most users would perform multiple different exercises in a single session. Fig. 17 highlights *RF-HAC*'s ability to accurately segment and classify an incoming RF point cloud stream with multiple exercises, where a subject performs 3 repetitions of lunges followed by 5 repetitions of squats.

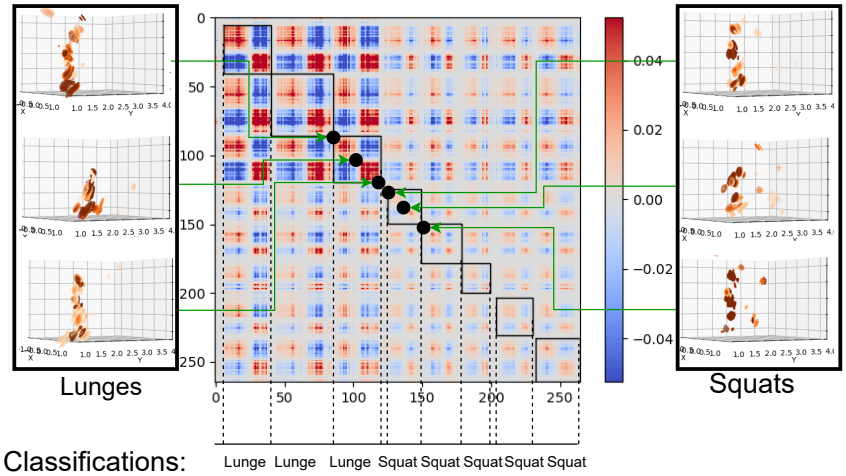


Fig. 17. Exercise mixture output

7 DISCUSSIONS AND CONCLUSIONS

RF-HAC is one of the first systems to bring automated and robust exercise cataloging to practice with privacy-preserving mmWave RF signals, where subjects are allowed to exercise in their natural pose/environment. It brings together innovations in self-supervised, pose-agnostic classification and doppler self-similarity based segmentation on unstructured 3D RF point clouds to realize this vision.

RF-HAC's robustness can be further enhanced in two directions: (i) exploring radar tilt angles (traditionally perpendicular to ground) to alleviate sparsity of point clouds for challenging exercises that are performed close to the ground (e.g. push-ups); and (ii) leveraging innovations in passive intelligent surfaces to provide additional diversity during deployment to tackle challenging NLoS scenarios.

REFERENCES

[1] [n. d.]. Intel® NUC 11 Enthusiast Mini PC - NUC11PHKi7CAA. https://www.intel.com/content/www/us/en/products/sku/195961/intel-nuc-11-enthusiast-mini-pc-nuc11phki7caa/specifications.html.
[2] [n. d.]. Linux cgroups. https://man7.org/linux/man-pages/man7/cgroups.7.html.
[3] [n. d.]. Review: Amazon Halo Rise. https://www.wired.com/review/amazon-halo-rise/.
[4] [n. d.]. RF-HAC datasets and code. https://bit.ly/493DXQw.
[5] [n. d.]. RF-HAC github. https://github.com/Ohesachite/radar-nn-model.
[6] [n. d.]. TI 3x4 radar chips. https://www.ti.com/product/AWR6843?utm_source=google&utm_medium=cpc&utm_campaign=epd-null-null-GPN_EN-cpc-pf-google-wwe&utm_content=AWR6843&ds_k=AWR6843&DCM=yes&gclid=CjwKCAjwoqGnBhAcEiwAwK-OkdCizqhgLwaVF_urOlSGw9HlM3UdDdlVbRSW7tEIcjmomfrNwm3xOBoCK4oQAvD_BwE&gclsrc=aw.ds.
[7] Aakriti Adhikari, Hem Regmi, Sanjib Sur, and Srihari Nelakuditi. 2022. MiShape: Accurate Human Silhouettes and Body Joints from Commodity Millimeter-Wave Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–31.
[8] Adeel Ahmad, June Chul Roh, Dan Wang, and Aish Dubey. 2018. Vital signs monitoring of multiple people using a FMCW millimeter-wave sensor. In *2018 IEEE Radar Conference (RadarConf18)*. IEEE, 1450–1455.
[9] Mubarak A Alanazi, Abdullah K Alhazmi, Osama Alsattam, Kara Gnau, Meghan Brown, Shannon Thiel, Kurt Jackson, and Vamsy P Chodavarapu. 2022. Towards a low-cost solution for gait analysis using millimeter wave sensor and machine learning. *Sensors* 22, 15 (2022), 5470.
[10] Sizhe An and Umit Y Ogras. 2021. Mars: mmwave-based assistive rehabilitation system for smart healthcare. *ACM Transactions on Embedded Computing Systems (TECS)* 20, 5s (2021), 1–22.
[11] Robert Bodor, Bennett Jackson, and Nikolaos Papanikolopoulos. 2003. Vision-based human tracking and activity recognition. In *Proc. of the 11th Mediterranean Conf. on Control and Automation*, Vol. 1. Citeseer, 1–6.

[12] Baicheng Chen, Huining Li, Zhengxiong Li, Xingyu Chen, Chenhan Xu, and Wenyao Xu. 2020. ThermoWave: a new paradigm of wireless passive temperature monitoring via mmWave sensing. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[13] L Minh Dang, Kyungbok Min, Hanxiang Wang, Md Jalil Piran, Cheol Hee Lee, and Hyeonjoon Moon. 2020. Sensor-based and vision-based human activity recognition: A comprehensive survey. *Pattern Recognition* 108 (2020), 107561.

[14] Ashutosh Dhekne, Mahanth Gowda, Yixuan Zhao, Haitham Hassanieh, and Romit Roy Choudhury. 2018. Liquid: A wireless liquid identifier. In *Proceedings of the 16th annual international conference on mobile systems, applications, and services*. 442–454.

[15] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. 12, null (jul 2011), 2121–2159.

[16] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. 2020. Counting Out Time: Class Agnostic Video Repetition Counting in the Wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[17] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (Portland, Oregon) *(KDD'96)*. AAAI Press, 226–231.

[18] Hehe Fan, Yi Yang, and Mohan Kankanhalli. 2021. Point 4D Transformer Networks for Spatio-Temporal Modeling in Point Cloud Videos. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*.

[19] Tianbo Gu, Zheng Fang, Zhicheng Yang, Pengfei Hu, and Prasant Mohapatra. 2019. Mmsense: Multi-person detection and identification via mmwave sensing. In *Proceedings of the 3rd ACM Workshop on Millimeter-wave Networks and Sensing Systems*. 45–50.

[20] Junfeng Guan, Sohrab Madani, Suraj Jog, Saurabh Gupta, and Haitham Hassanieh. 2020. Through fog high-resolution imaging using millimeter wave radar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11464–11473.

[21] Unsoo Ha, Salah Assana, and Fadel Adib. 2020. Contactless Seismocardiography via Deep Learning Radars. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking* (London, United Kingdom) *(MobiCom '20)*. Association for Computing Machinery, New York, NY, USA, Article 62, 14 pages. https://doi.org/10.1145/3372224.3419982

[22] Dan Hendrycks and Kevin Gimpel. 2023. Gaussian Error Linear Units (GELUs). arXiv:1606.08415 [cs.LG]

[23] Yash Jain, Chi Ian Tang, Chulhong Min, Fahim Kawsar, and Akhil Mathur. 2022. ColloSSL: Collaborative Self-Supervised Learning for Human Activity Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 1, Article 17 (mar 2022), 28 pages. https://doi.org/10.1145/3517246

[24] Sorachi Kato, Tomoki Murakami, Takuya Fujihashi, Takashi Watanabe, and Shunsuke Saruwatari. 2022. CBR-ACE: Counting human exercise using Wi-Fi beamforming reports. *Journal of Information Processing* 30 (2022), 66–74.

[25] Hao Kong, Xiangyu Xu, Jiadi Yu, Qilin Chen, Chenguang Ma, Yingying Chen, Yi-Chao Chen, and Linghe Kong. 2022. M3Track: <u>mm</U>wave-Based <u>m</U>ulti-User 3D Posture Tracking *(MobiSys '22)*. Association for Computing Machinery, New York, NY, USA, 491–503. https://doi.org/10.1145/3498361.3538926

[26] Der-Tsai Lee and Bruce J Schachter. 1980. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences* 9, 3 (1980), 219–242.

[27] Ofir Levy and Lior Wolf. 2015. Live repetition counting. In *Proceedings of the IEEE international conference on computer vision*. 3020–3028.

[28] Zhengxiong Li, Fenglong Ma, Aditya Singh Rathore, Zhuolin Yang, Baicheng Chen, Lu Su, and Wenyao Xu. 2020. Wavespy: Remote and through-wall screen attack via mmwave sensing. In *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 217–232.

[29] J. Lin. 1991. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory* 37, 1 (1991), 145–151. https://doi.org/10.1109/18.61115

[30] Haipeng Liu, Kening Cui, Kaiyuan Hu, Yuheng Wang, Anfu Zhou, Liang Liu, and Huadong Ma. 2022. mTransSee: Enabling Environment-Independent mmWave Sensing Based Gesture Recognition via Transfer Learning. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 1, Article 23 (mar 2022), 28 pages. https://doi.org/10.1145/3517231

[31] Hankai Liu, Xiulong Liu, Xin Xie, Xinyu Tong, and Keqiu Li. 2024. PmTrack: Enabling Personalized mmWave-based Human Tracking. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 4, Article 167 (jan 2024), 30 pages. https://doi.org/10.1145/3631433

[32] Lingfei Mo, Fan Li, Yanjia Zhu, and Anjie Huang. 2016. Human physical activity recognition based on computer vision with deep learning model. In *2016 IEEE international instrumentation and measurement technology conference proceedings*. IEEE, 1–6.

[33] Cecily Morrison, Peter Culmer, Helena Mentis, and Tamar Pincus. 2016. Vision-based body tracking: turning Kinect into a clinical tool. *Disability and Rehabilitation: Assistive Technology* 11, 6 (2016), 516–520.

[34] Herbert Robbins and Sutton Monro. 1951. A Stochastic Approximation Method. *The Annals of Mathematical Statistics* 22, 3 (1951), 400 – 407. https://doi.org/10.1214/aoms/1177729586

[35] Dariush Salami, Ramin Hasibi, Sameera Palipana, Petar Popovski, Tom Michoel, and Stephan Sigg. 2021. Tesla-Rapture: A Lightweight Gesture Recognition System From mmWave Radar Sparse Point Clouds. *IEEE Transactions on Mobile Computing* 22 (2021), 4946–4960. https://api.semanticscholar.org/CorpusID:247092918

[36] David Schneider, Saquib Sarfraz, Alina Roitberg, and Rainer Stiefelhagen. 2022. Pose-based contrastive learning for domain agnostic activity representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3433–3443.

[37] Akash Deep Singh, Sandeep Singh Sandha, Luis Garcia, and Mani Srivastava. 2019. RadHAR: Human Activity Recognition from Point Clouds Generated through a Millimeter-Wave Radar. In *Proceedings of the 3rd ACM Workshop on Millimeter-Wave Networks and Sensing Systems* (Los Cabos, Mexico) *(mmNets'19)*. Association for Computing Machinery, New York, NY, USA, 51–56. https://doi.org/10.1145/3349624.3356768

[38] Edward M Sitar and Sanjib Sur. 2022. MilliFit: Millimeter-Wave Wireless Sensing Based At-Home Exercise Classification. In *2022 18th International Conference on Mobility, Sensing and Networking (MSN)*. IEEE, 150–154.

[39] Elahe Soltanaghaei, Akarsh Prabhakara, Artur Balanuta, Matthew Anderson, Jan M Rabaey, Swarun Kumar, and Anthony Rowe. 2021. Millimetro: mmWave retro-reflective tags for accurate, long range localization. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 69–82.

[40] Girish Tiwari and Shalabh Gupta. 2021. An mmWave radar based real-time contactless fitness tracker using deep CNNs. *IEEE Sensors Journal* 21, 15 (2021), 17262–17270.

[41] Chao Wang, Feng Lin, Tiantian Liu, Kaidi Zheng, Zhibo Wang, Zhengxiong Li, Ming-Chun Huang, Wenyao Xu, and Kui Ren. 2022. mmEve: eavesdropping on smartphone's earpiece via COTS mmWave device. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 338–351.

[42] Yuheng Wang, Haipeng Liu, Kening Cui, Anfu Zhou, Wensheng Li, and Huadong Ma. 2021. m-Activity: Accurate and Real-Time Human Activity Recognition Via Millimeter Wave Radar. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 8298–8302. https://doi.org/10.1109/ICASSP39728.2021.9414686

[43] Yong Wang, Wen Wang, Mu Zhou, Aihu Ren, and Zengshan Tian. 2020. Remote monitoring of human vital signs based on 77-GHz mm-wave FMCW radar. *Sensors* 20, 10 (2020), 2999.

[44] Jingmiao Wu, Jie Wang, Qinghua Gao, Miao Pan, and Haixia Zhang. 2021. Path-independent device-free gait recognition using mmwave signals. *IEEE Transactions on Vehicular Technology* 70, 11 (2021), 11582–11592.

[45] Yang Xiao, Yuming Du, and Renaud Marlet. 2021. PoseContrast: Class-agnostic object viewpoint estimation in the wild with pose-aware contrastive learning. In *2021 International Conference on 3D Vision (3DV)*. IEEE, 74–84.

[46] Yucheng Xie, Ruizhe Jiang, Xiaonan Guo, Yan Wang, Jerry Cheng, and Yingying Chen. 2022. mmFit: Low-Effort Personalized Fitness Monitoring Using Millimeter Wave. In *2022 International Conference on Computer Communications and Networks (ICCCN)*. 1–10. https://doi.org/10.1109/ICCCN54977.2022.9868878

[47] Satyapreet Singh Yadav, Radha Agarwal, Kola Bharath, Sandeep Rao, and Chetan Singh Thakur. 2023. tinyRadar for Fitness: A Contactless Framework for Edge Computing. *IEEE Transactions on Biomedical Circuits and Systems* (2023).

[48] Jie Yan, Xianlin Zeng, Anfu Zhou, and Huadong Ma. 2022. MM-HAT: Transformer for Millimeter-Wave Sensing Based Human Activity Recognition. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. 547–553. https://doi.org/10.1109/GLOBECOM48099.2022.10000673

[49] Zhicheng Yang, Parth H Pathak, Yunze Zeng, Xixi Liran, and Prasant Mohapatra. 2016. Monitoring vital signs using millimeter wave. In *Proceedings of the 17th ACM international symposium on mobile ad hoc networking and computing*. 211–220.

[50] Chengxi Yu, Zhezhuang Xu, Kun Yan, Ying-Ren Chien, Shih-Hau Fang, and Hsiao-Chun Wu. 2022. Noninvasive Human Activity Recognition Using Millimeter-Wave Radar. *IEEE Systems Journal* 16, 2 (2022), 3036–3047. https://doi.org/10.1109/JSYST.2022.3140546

[51] Huaidong Zhang, Xuemiao Xu, Guoqiang Han, and Shengfeng He. 2020. Context-aware and scale-insensitive temporal repetition counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 670–678.

[52] Long Zhao, Yuxiao Wang, Jiaping Zhao, Liangzhe Yuan, Jennifer J. Sun, Florian Schroff, Hartwig Adam, Xi Peng, Dimitris Metaxas, and Ting Liu. 2021. Learning View-Disentangled Human Pose Representation by Contrastive Cross-View Mutual Information Maximization. arXiv:2012.01405 [cs.CV]

[53] Tao Zhou, Zhaoyang Xia, Xiangfeng Wang, and Feng Xu. 2021. Human sleep posture recognition based on millimeter-wave radar. In *2021 Signal Processing Symposium (SPSympo)*. IEEE, 316–321.